The use of User Centred Design methodologies in mobile application design: A literature review.

Thabo Kopane.
UCT Computer Science Department.
Email: kpntha001@myuct.ac.za

Abstract

The purpose of this article is to review the available literature regarding the use of User-Centred Design or Human-Centred Design methodologies in software engineering problems or software engineering research problems. We wanted to find if there have been attempted use of UCD in research and whether it has worked, and we try to find reasons why some projects opt not to include users in their project.

The goal is to analyse the most available papers we most possibly can that involve the creation of an app for parking or navigation and if some of those studies were done on a University campus setting, that would be even better. In general, we want to see the design or development methodologies used for companion applications and whether there is a specific method within UCD that many making companion apps seem to gravitate to.

Introduction

There are a lot of devices that make fall under the internet of things group; smartwatches, smart lights, home sensors and many other tech devices we use. Most of these devices have a companion app, mainly because of how ubiquitous the smartphone has become over the last decade.

The problem that we want to solve in our project is the issue of parking availability on campus and having near real-time access to parking information. For our potential solution, we propose to design a Arduino based sensor that sends information to a server and that information would be displayed on a smartphone-based application, to aid users in finding available parking in any of the campus parking bays that we may have put our sensors on.

Pollak and Kattak[1993] point out that it is beneficial to users to know information relating to parking ahead of time, and that it could potentially prevent the stress associated with parking or the problems that drivers may often encounter with other drivers. One of the main drivers behind our projects is that we want to make sure that drivers (both students and academic staff) have prior information relating to available parking on campus and that finding parking is as efficient as possible.

There are many different apps discussed in literature, relating to parking apps on a university campus; some focus on end-user crowdsourced information about available parking(Kopecky and Domingue, 2012), some are about integrating apps with National Geopark tours, hikes or something (Kisser and Zecha, 2017), and some apps use CCTV cameras installed in

parking bays to help users find parking(Hasenfuss and Fraifer et al, 2018).

Steen et al[2007] show, through a review of different research studies on design and software design, the benefits of involving users early on in the development process.
Through this review, we wish to find answers to the following questions: Are there smart parking application technologies on university campuses using UCD/HCD methodologies? Do researchers designing apps use UCD? What are the benefits of using UCD/HCD methodologies for app design? Can we use some of the features of the smart parking apps, even when they aren't for university campuses?

Features

The literature includes a lot of useful and interesting features for the design of parking apps, some parking apps are designed to be commercialized and thus certain features make sense in those apps. Finding available parking is often akin to wayfinding or navigation around a specific area, most of the apps use a map – but different maps for different reasons. The Openstreet map API is used because it is open source (Kopecky and Domingue, 2012), and so for the creators of ParkJam – which is a crowdsourcing-based parking app, a map where users can add missing information was a sensible choice.
ParkJam, in addition to allowing end-users to submit information, has a reliable data source feature, where information from carpark operators is weighted more than general information from other end users.
Google maps doesn't have available tools for navigating between buildings (Cardei and Raviv et al, 2013) and therefore for the *Campus Assistant Application,* they made a map data structure, called MapEditor, that takes into account the geography on campus, to help users find their way around campus, the map still makes use of google maps for terrain and other things, it is an added compliment to make google maps useful for the app's context. Users can add or edit information on the MapEditor. The app also uses popular algorithms for navigation, no wheel needed to be invented here.
Some map functionalities allow users, with the aid of a web application – in addition to a smartphone app, to draw areas on maps to define a parking location (Amoretti and Picone et al, 2013). This feature was seen to be useful for parking operators to define where they work, and the app has an algorithm to prevent users (parking operators) from defining an area where another user defined an area.
Having an option for different users seems to be a universal design for all the apps, Amoretti et al takes it further by having standalone apps for parking operators and for parking users. Users of the National Geoparks apps also suggested standalone apps for different users.

For most parking solutions mentioned in literature, the goal was for that solution to be used in city or a town – so one included feature in most, if not all, of the apps is a billing information feature that is either automatic (Yang et al,2012) or a carpark operator initiates the parking session (Amoretti and Picone et al, 2013).

The literature review focuses on apps in general, with a particular interest on parking apps. The following are useful features from non-parking apps.
Having notable landmarks to let users know where they are according to the map they are following (Kisser and Zecha, 2017).
Allowing users to make appointments through the app (Smolenaers et al, 2018).
Posti et al(2014) made an app for asocial

hikers, one of the features was being able to detect the most used path using images on social media sites such as Flickr and information from googlep maps. Another potentially useful feature for wayfinding purposes is route planning (Magrini, Nati and Panizzi, 2012)

Design Apps

The involvement of users during the design phases, is the basic idea behind User Centred Design (Steen, 2007). ParkJam, previously mentioned, is an app that crowdsources parking information from other end-users (Kopecky and Domigue, 2012). The authors didn't include end-users in the design phase or testing phase, the literature doesn't mention usability testing. It is often thought that involvement of users in design phase may stymie or impact on innovation, but sometimes designers create items that don't solve the problem or solve the problem in a way that the user is unhappy with [Steen et al, 2007]

UCD methodologies can be applied specifically for the design of a smartphone application or for a general problem, like the problem of finding parking. Hasenfuss et al's research paper focused on the use of UCD methodologies not just for the app, for the parking solution. Users were involved from the design phase; they were interviewed during iterations and focus groups involving users were ran before the final design of both the IoT parking system and the app. The idea wasn't just to involve users when testing the final product but consult users on added features and interactions with the system and the apps. (Hasenfuss and Fraifer et al, 2013). The solution had positive feedback from end-users, most users thought it thoroughly solved the problem, it was fit for the users and there was no invention of users.

Some National Geoparks want to be able to integrate apps within their parks, have tourists and students and hikers and teachers be able to use the app for wayfinding, learning and also for the added experience of visiting the park (Kissed and Zecha, 2017). The researcher therefore wanted to find out the current experiences of users in geoparks with apps and geoparks without apps – the focus for us is for the geoparks with apps. Using UCD methodologies, another park's app was assessed – The *National Geopark Bergstrasse Odenwald* app.
Users were interviewed on the app and there was also a focus group for different groups of users for the app. The *National Geopark Bergstrasse Odenwald* app was not designed using User Centred Design methodologies.
Most of the users during the interview and the focus group were mostly disappointed by the app, they found that it wasn't offering anything worthwhile for them as users. Users said that it functioned as a map for the most part but provided nothing educational. One of the reasons the app was designed the way because the city had made a wayfinding app for the city and thus it was cheaper to use the available technology to make a similar app, thus the creation didn't involve users as already stated and users didn't like the app overall.
The users made suggestions for features if the researchers wished to make an app. Adding short videos, animated features – to make the app engaging. Having as little text as possible, quick information look-up and have an app that functions as an assistant for the park tour and not distract.
The app, some users felt, should enhance the experience of the geopark; users said

that since network doesn't always work well in remote areas, an app that doesn't depend on the internet would be good. Some users also felt that a mini tutorial for new users opening the app should be included.
A lot of users said that countless people visit the park for different reasons and suggested that designers should make different apps for different users – or have prompts asking what's the purpose of users visits in order to tailor an experience that users would enjoy.

Another app with wayfinding capabilities that was created using the UCD methodologies is an *Asocial Hiking App (Posti and Schoning et al, 2014)*. The idea of the app, according to researchers, is to help users who prefer solitary hiking to finding routes that are usually empty for the "asocial" users to hike.
For the research on potential users of the app, an online survey was circulated to find out about the demographics and behaviour of hikers and whether hikers use map apps to plan their routes and on their routes. The results showed that most hikers said that they used map-based application around 90% of the time, but rarely tracked their hikes.
This is a pre-design survey; users were split on the asocial hiking concept – some solitary hikers weren't averse to encountering other hikers on any trail routes that they chose.
The design of the app focused a lot on involving users throughout, regardless of whether users some users didn't understand the point of the app – some said that during the conception phase that the app was trying to solve a problem that didn't exist – the users were involved on the design choice, the interaction of the app, notifications that the app sends out and other features that were added to the app. When the final prototype was created, the researchers ran a focus group to test the usability of the app – and the researcher also ran an experiment with 8 potential users to go a hike and to see whether they'd use the app or not.
The most important aspects of this paper were the focus it had on UCD methodologies for the design of the app, we aren't sure whether the app wound up being used by hikers seeking solitary routes.

Another app that we didn't delve in was a Hospital wayfinding app (Smolenaers et al, 2019). The idea of the app for users to be able to find their way around hospital wards efficiently – the researchers used UCD methodologies for the app design, but it wasn't as robust as the hiking app or the geopark app.

UCD as a software engineering method

User Centred Design came out of Postmodernism (Redstrom, 2005). In the design of objects in the postmodernist era, designers were moving away from designing the form to designing for function over form, *form follows function*. Before this transformation where form is subordinate to function(Sullivan, 1986), designers did not take users into account when designing objects and during that time, it was felt that it was necessary for innovation to be the benevolent dictator when it comes to users' needs.(Panne et al, 2003).
Designers wanted to create objects that solve problems users were facing, and at the time it didn't particularly matter whether the solution will create a misfit or not because It was seen as solving the problem.
Hackos and Redish(1998) were quoted in Redstrom(2005) saying " we have to appreciate that we cannot just impose

designs on people", they went on to say that unlike computer and computer machinery, humans are unpredictable and have to be studied in order to find out what kind of design would fit for their usage. One of the early processes of deigns was to knowingly or unknowingly invent users, many designers had scenarios that they thought objects would be used in, and thus designed solutions for those scenarios. (Steen et al, 2007).

Involving users in design or development phase is trying to find out how users will interact with the end systems. A lot of companies implement agile development methodologies and find it difficult to include aspects of UCD methodologies (Mclnerney and Maurer, 2004). Maurer and Mclnerney (2004) go on to say that the focus in agile development is on the developers working with customers, adding a representative of users to the team is often seen as adding a layer of complexity.
We focus on agile development because it is one of the most widely used development methodology in the industry. Agile development focuses on how the software should be made and UCD is a process that focusses on the needs and requirement of users when it comes to using an object or software product (Sohaib and Khan, 2010). The focus on customer feedback means that a software is made iteratively like how agile development does iterative software design but include user testing during that iterative development.
User testing adds more time to the development of the software and it often costs an additional amount of money – thus many development teams are apprehensive about incorporating elements of UCD into their development (Sohaib and Khan, 2010).

Projects that involve users take longer on average, evidence shows that that time it takes is worth it and that on average users give positive feedback to a software solution that involves users (Hasenfuss and Fraifer et al, 2018).

Conclusions

Literature for smart parking solutions do not focus on the usability design of an application if it is included. When app creators include elements of interface design, users aren't included or involved with the design, the majority of apps created don't follow UCD methodologies and some of the reasons include time, the added cost or the perceived added complexities of involving users in the development phase.
With regards to the problem of money, most researchers' project doesn't seem to get a lot of funding and the projects are mostly done as a proof of concept.

The ParkJam app mentioned earlier, was a crowdsourcing app, it required user input to populate the app with information of available parking. No users were involved in the design of the app and the paper doesn't mention any usability testing of the final product either. For an app that required user input, without incentives, it was very odd that it didn't involve users. Domingue and Kopecky, when reached out for comment said that the app was a failure, reasons could've been lack of users due to bad fit and non-user centric design.
Integrating the app with google maps API, including tiered users for academic staff and students, ability to add missing information on the app or map similarly to the crowdsourcing features are few potentially useful features that we can make use of for our proposed project.
UCD methodologies in the IoT space are very young and rarely used, UCD in

software engineering is not used as much as it should for a slew of reasons. Apps that make use of UCD benefit by hearing the input and feedback of users/customers during pre-production and a software development is structure properly, there might not be a delay, rather than a delay happening because users hate features when a software is deployed.

Incorporating UCD in software design or software engineering is possible and has produced positive results, example would be the papers written by Kisser and Zecha, Hasenfuss and Fraifer and Posti and Schoning.

References

Cardei M, Zankina I, Cardei I, et al. 2013. Campus Assistant Application on an Android Platform. 2013 Proceedings of IEEE Southeastcon. Jacksonville, Florida. 1-6.

Grazioli A, Amoretti M, Picone M, et al. 2013. Collaborative Mobile Application and Advanced Services for Smart Parking. 2013 IEEE 14th International Conference on Mobile Data Management. Milan, Italy. 39 – 44

Hasenfuss H, Fraifer M, Kharel S, et al. 2018. It takes two to Tango: Merging Science and Creativity to Support Continued Innovation in the IoT Domain. Advances in Science, Technology and Engineering Systems Journal Vol 3, No. 5. Ireland. 82-91

Kisser T & Zecha S. 2017. Creating effective Mobile Apps for National Geoparks. An explorative Study. GI Forum Vol 5, Issue 2. Germany. 113-125

Kopecký J, Domingue J. 2015. ParkJamJAM: Crowdsourcing Parking Availability Information with Linked Data. The Semantic Web: ESWC 2012 Satellite Events. ESWC 2012. Lecture Notes in Computer Science, Vol 7540. Springer, Berlin, Heidelberg. 381 – 386.

Fraifer M & Fernstrom M. 2017. Designing a smart car parking system (PoC) Prototype Utilizing CCTV Nodes: A vision of an IoT parking system via UCD process. Advances in Science, Technology and Engineering Systems Journal Vol 3, No. 3. Ireland. 755-764.

Rdestrom J. 2005. Towards user design? On the shift from object to user as the subject of design. Design studies Vol 27. Gotenborg, Sweden. pp. 123-139.

Khattak A & Polak J. 1993. Effect of parking information on travelers' knowledge and behavior. Transportation vol 20. Oxford, UK. 373-393.

Steen M, Klok J, Kuijt-Evers L. 2007. Early User Involvement in Research and Design Projects – A review of methods and practices. 23rd EGOS colloquium: Dancing with users – How to organize innovation with consumers and users? Vienna, Austria.

Lin T, Rivano H, Mouel F. 2017. A Survey of smart parking solutions. IEEE Transactions on Intelligent Transportation Systems. Pp. 3229-3253

Sohaib O, Khan K. 2010. Integrating usability engineering and agile software development: A literature review. 2010 International Conference on Computer Design and Applications. Quinhuangdao, China. V2.32 – V2.38.

Mclnerny P, Maurer F. 2005. UCD in agile projects: Dream team or odd couple? Interactions. Calgary. 19-23

Mahmud F, Aris H. 2015. State of mobile crowdsourcing applications: A review. 2015 4th International Conference on Software Engineering and Computer Systems (ICSECS). Malaysia. 27- 32

Posti M, Schoning J, Hakkila J. 2014. Unexpected Jouneys with the HOBBIT - The design and evaluation of an asocial hiking app. DIS 2014. Vancouver , Canada. 637 – 646.

Kujala Sari. 2003. User involvement: A review of benefits and challenges. Behaviour & Information Technology. 01 January 2003. 1 – 16.

Smolenaers F, Chestney T, Walsh J, et al. 2019. User Centred Development of a Smartphone Application for Wayfinding in a Complex Hospital Environment. 20th Conference of the International Ergonomics Association. Switzerland. 383-393.